

Using distributed version control systems to improve project management and collaboration

Benjamin Chauvette

Rice University

chauvette@rice.edu

Outline

- Why look to software development?
- Local and centralized version control
- Distributed version control
 - Cloning
 - Branching & Merging
 - Network independence
- Challenges with distributed version control
- Which version control system to use?

Why look to software development?

- Similar challenges
- Similar filetypes
- Similar need for file histories

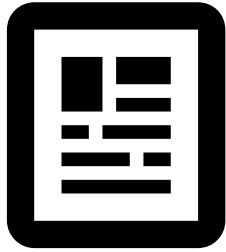
Because it works!



What is a version control system (VCS)?

Software that tracks
changes to files over time

VCS: Basic workflow

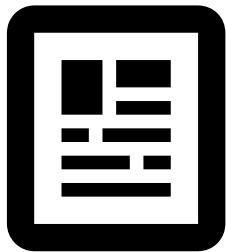


Working copy



Work

Edit and save your files



Changed copy

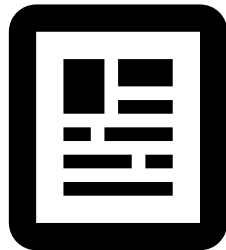


Stage

Tell VCS which changes to notice



Old Version



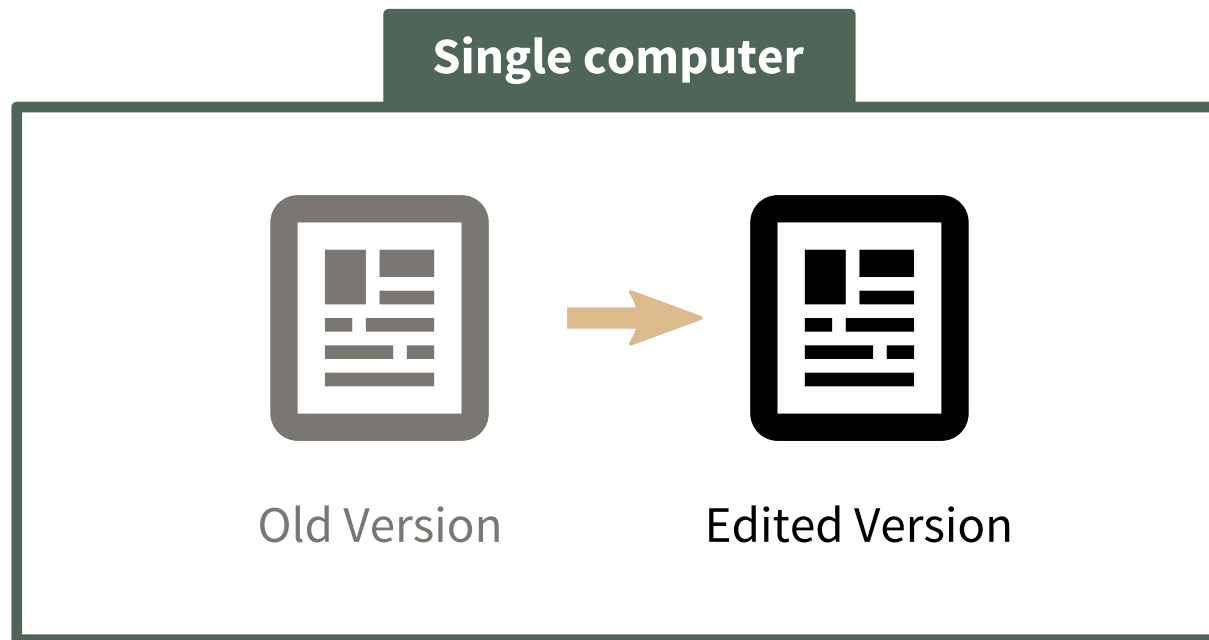
Edited Version



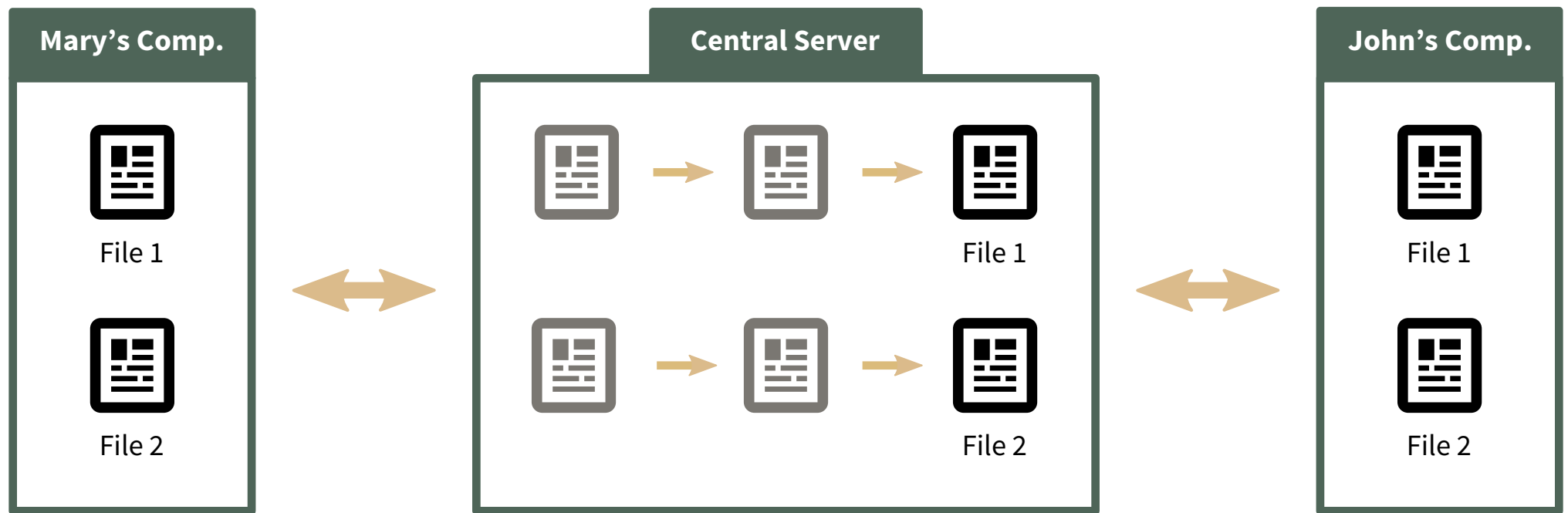
Commit

Enter changes into a repository

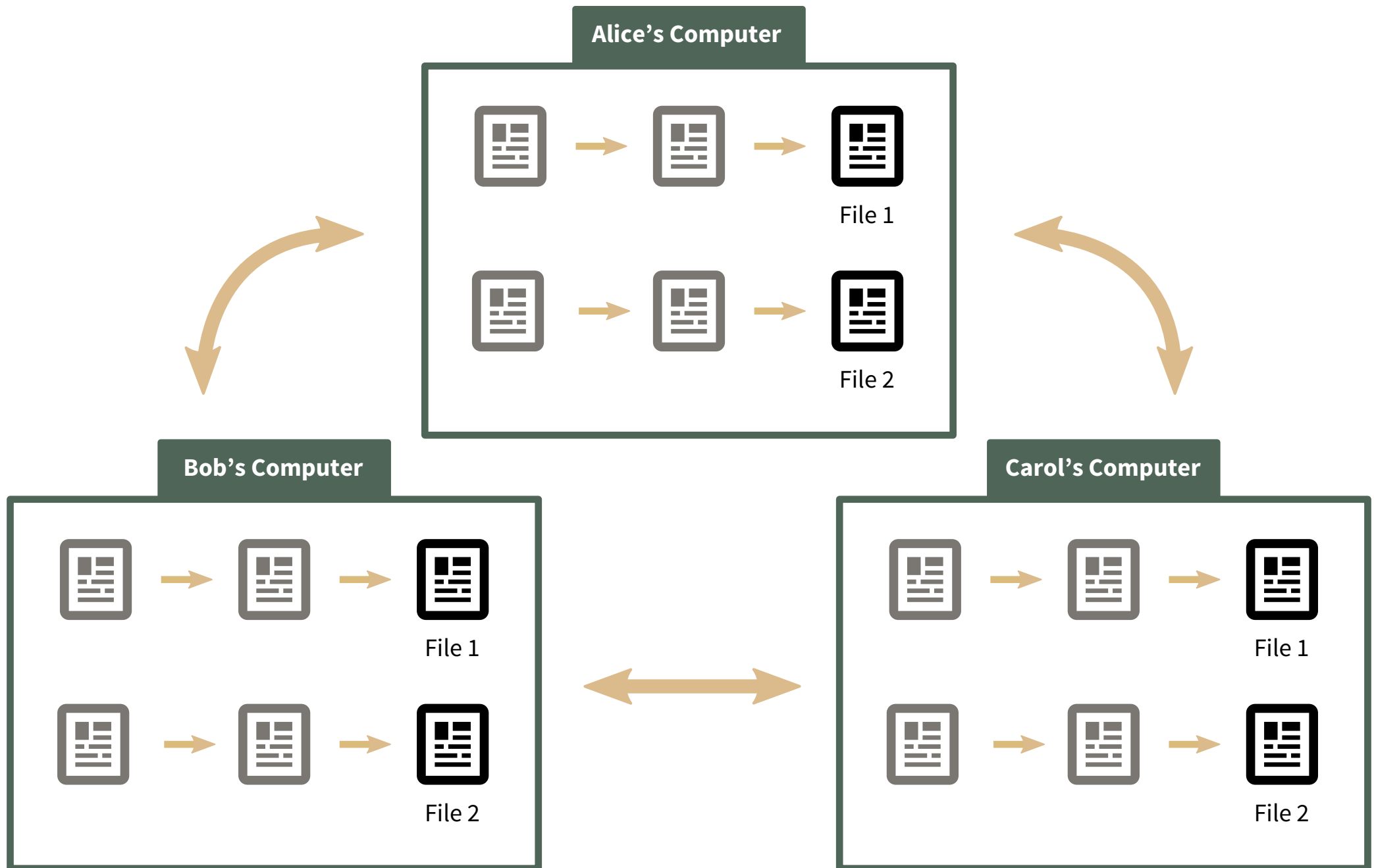
Local VCS



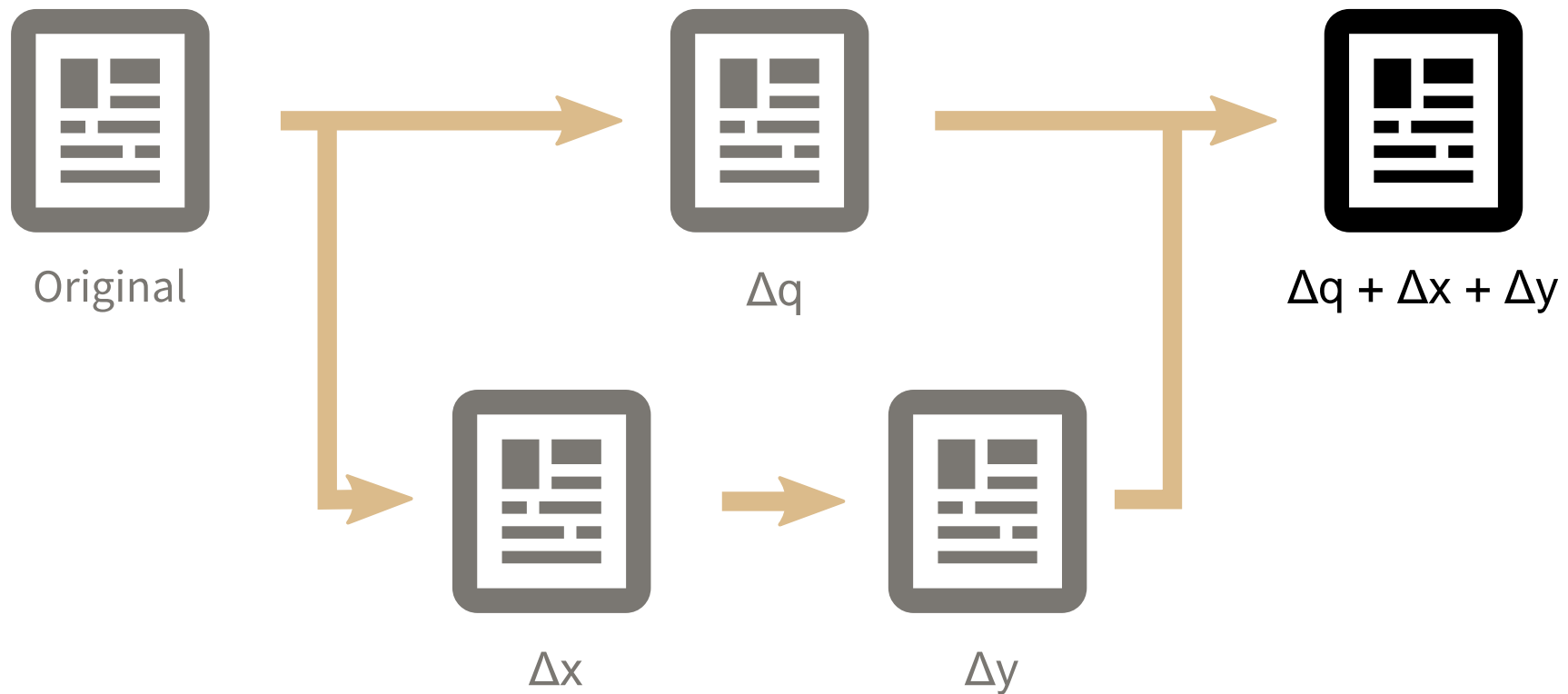
Centralized VCS



Distributed VCS: Cloning



Distributed VCS: Branching & Merging



Distributed VCS: Network independence



CDs,
DVDs



Hard drives,
USB drives

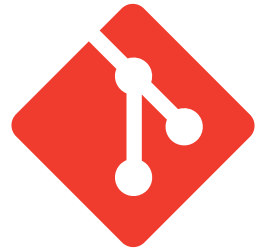


Internet,
Local networks

Distributed VCS: Challenges

- Steep learning curve
- Merging XML, Toolbox files
- Developing new workflows, best practices

So, which system to use?



git

- Extremely flexible
- Great tutorials
- Infamously difficult



mercurial

- Secure history
- Less complicated
- Fewer tutorials

Chorus

- Designed for (non-)linguists
- Least complicated
- Very little documentation

Conclusion

DVCS can help satisfy the need
for more collaborative workflows

We should develop DVCS-based workflows
and establish best-practices to fully
realize their potential

References & Resources

- **Git:** <http://www.git-scm.com>
- **Mercurial:** <http://mercurial.selenic.com>
- **Chorus:** <http://projects.palaso.org/projects/chorus>
- **Git tutorials:** <http://sixrevisions.com/resources/git-tutorials-beginners/>
- **Mercurial tutorials:** <http://mercurial.selenic.com/wiki/BeginnersGuides>
- **SourceTree:** <http://www.sourcetreeapp.com/>
Graphical program for managing Git and Mercurial repositories
- Chacon, S. 2009. *Pro Git*. New York: Apress.
Online version: <http://git-scm.com/book>
- O'sullivan, B. 2009. *Mercurial: the definitive guide*. Sebastopol, CA: O'Reilly.
Online version: <http://hgbook.red-bean.com>
- Hatton, J. 2011. Software for remote dictionary Collaboration.
Intl J Lexicography 24(4). 420-431. DOI: [10.1093/ijl/ecr020](https://doi.org/10.1093/ijl/ecr020)